Docket No.: AT9-98-737

PATENT

#10
LDT
9-5-02

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Chavez, Jr.**

Serial No.: **09/292,190**

Group Art Unit: **2155**

Filed: **April 15, 1999**

Examiner: **Dinh, Khanh Q.**

For: **Method and System for Enabling a Network Function in a Context of One or All Server Names in a Multiple Server Name Environment**

Assistant Commissioner for Patents
Washington, D.C. 20231

ATTENTION: Board of Patent Appeals and Interferences

Certificate of Mailing Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being deposited with the United States Postal Service as First Class mail in an envelope addressed to: Assistant Commissioner of Patents, Washington, D.C. 20231 on August 23, 2002.

By: Dell Whitton
Dell Whitton

APPELLANT'S BRIEF (37 C.F.R. 1.192)

This brief is in furtherance of the Notice of Appeal, filed in this case on June 27, 2002.

The fees required under § 1.17(c), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate. (37 C.F.R. 1.192(a))

REAL PARTIES IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines, Inc.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-22

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: NONE
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 1-22
4. Claims allowed: NONE
5. Claims objected to: 8, 9, 11 and 18-20
6. Claims rejected: 1-7, 10, 12-17, 21 and 22

C. CLAIMS ON APPEAL

The claims on appeal are: 1-7, 10, 12-17, 21 and 22.

STATUS OF AMENDMENTS

No amendments have been filed after issuance of the Final Office Action. A Response to the Final Office Action was filed on March 8, 2002 containing only remarks. Based on the Response to the Final Office Action, the Examiner withdrew the rejections of

claims 8, 9, 11 and 18-20 and indicated these claims to contain allowable subject matter in the Advisory Action mailed June 14, 2002. However, for the reasons set forth herein below, Appellant respectfully submits that all of the claims are directed to allowable subject matter.

SUMMARY OF INVENTION

The present invention provides a method of executing a function on a server in a distributed data processing system. The server responds to requests directed to a set of server names. A function request has an input that specifies a server name in the set of server names. The function is executed on the server in a server name context specified by the input containing the server name. The server name context on the server has a set of resources associated with a server name. A unique server name tag is generated for each server name in the set of server names and each resource in the set of resources is identifiable by the server name tag associatively stored with the resource.

ISSUES

The only issue on appeal is whether claims 1-7, 10, 12-17, 21 and 22 are anticipated under 35 U.S.C. § 102(e) by Wolff et al. (U.S. Patent No. 6,044,367).

GROUPING OF CLAIMS

The claims do not stand or fall together. The grouping of claims is as follows:

Group I – claims 1, 7, 12 and 21;

Group II – claims 2, 13 and 22;

Group III – claims 3 and 14;

Group IV – claims 4 and 15;

Group V – claim 5 and 16;

Group VI – claim 6 and 17; and

Group VII – claim 10.

ARGUMENT

I. 35 U.S.C. § 102, Anticipation

The Final Office Action maintains the rejection of claims 1-7, 10, 12-17, 21 and 22 under 35 U.S.C. § 102(e) as being anticipated by Wolff et al. (U.S. Patent No. 6,044,367). This rejection is respectfully traversed.

As to independent claim 1, the Final Office Action states:

> As to claim 1, Wolff discloses a method for executing a function on a server in a distributed data processing system, the method comprising the computer-implemented steps of:
>
> receiving a request for a function (I/O functions), wherein the request comprises an input specifying a server name, wherein the server responds to requests directed to a set of server names (plurality of servers 104A-106A of fig. 1A), and executing the function in a server name context on the server as directed by the input specifying the server name (see abstract, Figs. 1A, 2A, 3A, col. 4 line 14 to col. 5 line 67, col. 6 line 31 to col. 7 line 58).

Claim 1, which is representative of the other independent claims 12 (system) and 21 (computer program product) with respect to similarly recited subject matter, reads as follows:

> 1. A method for executing a function on a server in a distributed data processing system, the method comprising the computer-implemented steps of:
>
> receiving a request for a function, wherein the request comprises an input specifying a server name, wherein the server responds to requests directed to a set of server names; and
>
> executing the function in a server name context on the server as directed by the input specifying the server name.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed.

Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983).

Appellant respectfully submits that every element of Appellant's claimed invention is not identically shown in the Wolff reference arranged as they are in the present claims. Specifically, the Wolff reference does not teach "...wherein the server responds to requests directed to a set of server names; and executing the function in a server name context on the server as directed by the input specifying the server name" (emphasis added), as recited in claim 1.

Wolff teaches a distributed I/O store in which clients may access a resource through different paths. The Wolff system provides for an "aware" client as shown in Figure 2B, that includes a name driver 194 that is utilized by a fail-over module 188 and a load balancer module 190. In the Wolff system, as servers and resources become active in the system, load balancing is performed to reallocate which servers handle which resources in the network. This load balancing is described in column 7, line 16 to column 8, line 24. The results of the load balancing are replicated to each server's copy of a dynamic RAM resident configuration database.

The name driver module 194 of the "aware" clients maintains "an abstraction mapping of the network namespace resources, and [combines] all available paths for each volume to each node as a single computing resource available for use by the rest of the system." In other words, the name driver module 194 provides a single system image that includes all of the possible paths for reaching the various resources on the system. The load balancer module 190 calls the name driver module 194 to remap future I/O while the fail-over module 188 calls the name driver module 194 to retry I/O on another path (column 12, lines 42-55).

The network namespace resources are presented in the network namespace as part of a logical namespace and a physical namespace. The logical namespace presents a persistent view of the resources on the network. The physical namespace presents the

individual physical connection points used at any particular time to service the logical resources.

Thus, the name driver module 194 is used as a mechanism for providing a single system image (SSI) identifying other nodes in the network to which requests may be directed (column 19, lines 25-32). That is, the name driver module 194 is used to determine other nodes to which requests will be remapped for future I/O by the load balancer module 190 and nodes along another path that may be tried by the fail-over module 188 (column 12, lines 43-48). The name driver module 194 does not include a set of server names for which the current server responds to requests. In actuality, the name driver module 194 of Wolff is the exact opposite and identifies other nodes to which requests should be directed, not which requests should be handled by a current server based on a set of server names which the current server responds to. In fact, each server in the SSI provided by the name driver module 194 only has a single identity. There is no provision in Wolff for servers to handle requests directed to a set of server names.

Furthermore, the name driver module 194 is not associated with a server to which requests are sent. To the contrary, the name driver module 194 is part of an "aware" client. The "aware" client is not a server and is not a server that executes a "function in a server name context on the server as directed by the input specifying the server name." Moreover, the servers that are taught by Wolff do not include a name driver module and do not have an associated set of server names. There is no mention that any of these servers have a set of server names identifying which requests the server responds to. There is further no mention that a server in Wolff executes a function in a server name context on the server as directed by an input specifying the server name. Since Wolff does not teach a set of server names, there is no reason to execute functions in a server name context on Wolff because there is no ability to switch context in the Wolff system. In other words, each server in the Wolff system only responds to a single server name and therefore, there is no need to maintain different server name contexts.

The Final Office Action seems to suggest, although the Final Office Action fails to explain what elements of Wolff it considers similar to the claimed features, that simply because Wolff mentions a namespace and a name driver module, that somehow this general teaching rises to the level of anticipating the specific feature of a server having a set of

server names to which it responds and executes functions in server name contexts. This leap in interpretation of Wolff is not supported by the actual teachings of Wolff. Rather, this interpretation is taken completely from hindsight reconstruction having had the benefit of Appellant's disclosure. The Examiner is reading in teachings to the Wolff reference that simply are not there.

In summary, as noted above, Wolff teaches the opposite of the claimed invention in that the name driver module, which is in the "aware" client, identifies other nodes to which requests should be routed based on the network namespace and does not identify a set of server names for which a current server responds to requests. Furthermore, Wolff does not teach executing a function in a server name context because neither the "aware" client nor the servers in Wolff execute any functions in a server name context. All that Wolff teaches is the ability to perform load balancing to change the allocation of resources to servers and a name driver module that identifies all of the current paths to the resources as a single system image from which "aware" clients may determine how to perform load balancing or recovery by identifying alternate paths to a resource. Nowhere in Wolff is it ever taught or even suggested that servers may handle requests directed to different server names or that servers may handle such requests in server name contexts. Furthermore, the Final Office Action has not pointed to any section of Wolff that would allegedly teach such features and thus, has not established a case of anticipation based on Wolff.

The above arguments were originally presented in the Response to the First Office Action filed November 26, 2001. In response to Appellant's arguments, the Final Office Action merely states:

> Examiner respectfully disagrees. Wolff clearly discloses the server responding to requests directed to a set of server names and executing the function in a server name context on the server as directed by the input specifying the server name (i.e., using some administrative servers such as 104B and 106B of fig. 1A to handle requests from normal client 100A, then the server 106B passes the I/O request via a path to the administrative server, see fig. 1A, 1B, 2A, col. 4 line 14 to col. 5 line 67 and col. 6 line 31 to col. 7 line 58).
>
> In response to Appellant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a

> reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the Appellant's disclosure, such a reconstruction is proper. See in re Mclaughlin, 443 F. 2d 1392, 170 USPQ 209 (CCPA 1971).
>
> Therefore, the examiner asserts that Wolff teaches or suggests the subject matter broadly recited in independent claims 1, 12 and 21 as required under 35 U.S.C. 102(e). Claims 2-11, 13-20 and 22 are rejected at least by virtue of their dependency on independent claims 1, 12 and 21 and by other reasons set forth in the previous Office Action (paper #3, mailed on 8/28/2001).

First, it should be noted that the present rejection is under 35 U.S.C. § 102(e) and not 35 U.S.C. § 103(a) and thus, any alleged "suggestions" are not relevant to the present rejection. The Examiner appears to be confused as to the standard required for a 35 U.S.C. § 102(e) rejection which is further evident in the fact that the reference cited does not teach anything in the current claims. As stated above, under 35 U.S.C. § 102(e), every element of the claim must be identically shown in the reference arranged as they are in the claim. If every element of the claim is not identically shown in the reference, the rejection must be withdrawn. Alleged "suggestions" have no place in a rejection under 35 U.S.C. § 102(e). Furthermore, Appellant respectfully submits that the Wolff reference does not provide any suggestion to modify the Wolff reference to arrive at Appellant's claimed invention as stated herein.

With regard to the Examiner's allegations regarding the teachings of Wolff, based on the Examiner's statements above, it appears that the Examiner believes that simply because the Wolff reference teaches communication with servers and the redirection of requests from a first server to an administrative server that may then redirect the request to another server based on load balancing criteria, that somehow this is the exact same thing as a server responding to requests directed to a set of server names and executing a function in a server name context on the server as directed by the input specifying the server name. Appellants respectfully disagree.

None of the servers described in Wolff respond to a set of server names. Quite the contrary - each server in Wolff responds to one and only one server name. While Wolff does teach redirection of requests to memory resources based on a load balancing

mechanism or determination that a server is not available, e.g., due to a failure, such redirection is not the same as a server responding to a set of server names. Rather, in the mechanism of Wolff, as clearly described in the very section cited by the Office Action (column 6, line 31 to column 7, line 58), a client makes an I/O request for a file system 122 on a memory resource 118, requests for this resource being handled by server 106B. The server 106B passes the request on to the administrative server 104B which then determines if the client has access privileges, determines if additional memory space needs to be allocated, and then returns a block list to the server 106B which then handles subsequent reads/writes to the blocks in the block list. In this way, the Wolff mechanism allows one server to handle the administrative tasks of managing a memory resource while other servers are allowed to handle the transfer of data to and from this memory resource (column 7, lines 2-7).

The remainder of columns 7 and 8 discuss the rebalancing of I/O and administrative tasks with regard to the various memory resources. Such rebalancing involves re-mapping which available memory resources are handled by which of the servers 104C-106C (column 7, lines 65-68). Thus, for example, at a first time t1, server 104C may handle memory blocks 1-200 of a memory resource and at a second time t2, these memory blocks may be handled by a different server 105C depending on the criteria listed in column 7, lines 60-65. The configuration database records are used as a means for identifying which of the servers 104C-106C currently handle the I/O requests to a particular memory resource.

At any point, however, the servers 104C-106C only respond to one name even though they may handle different blocks of the memory resources. For example, if server 104C is handling blocks 1-200 of a memory resource and a client device wishes to write to these blocks of the memory resource, the administrative server looks to the configuration database records and determines which server is handling these blocks and routes the request to that server. Server 104C still only answers to its own server name, e.g., "server 104C," and the administrative server must route the request to "server 104C." Server 104C does not automatically respond to any request sent to any one of a set of server names. That is, server 104C does not respond to requests directed to "server 104C," "server 104D" and server "104E", for example. There is nothing in Wolff that

describes a server being able to respond to requests directed to a set of server names, as recited in independent claims 1, 12 and 21.

Since all of the servers in the Wolff network only respond to their own server name and not a set of server names, the servers in the Wolff network do not execute a function in a server name context. There is no need, in the servers of Wolff, to execute functions in a server name context because each server only responds to one server name and thus, there is no ability to have a plurality of contexts. Since there is no ability to have a plurality of contexts, there is no need to distinguish between contexts based on a server name.

By responding to requests directed to a set of server names and executing functions in a server name context, the present invention allows a single server to act as if it were multiple separate machines. Such an ability is not found in the servers of Wolff because, no matter what remapping of the memory resources is performed, the server always acts as a single server device responding to requests routed to it by the administrative server using the server's one and only name. There is nothing in Wolff that provides a single server the ability to respond to a set of server names.

In addition, the Examiner responds to Appellant's assertion that the Examiner is engaging in hindsight reconstruction by merely including a form paragraph in his response that states in part "But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the Appellant's disclosure, such a reconstruction is proper. See in re Mclaughlin, 443 F. 2d 1392, 170 USPQ 209 (CCPA 1971)" (emphasis added). Appellant's very argument is that the Examiner did include knowledge gleaned only from Appellant's disclosure. There is nothing remotely similar to Appellant's claimed invention in the Wolff reference. Wolff is directed to a system for load balancing to determine which servers should handle I/O requests to certain memory resources whereas the presently claimed invention is directed to a mechanism in which a server responds to a set of server names and executes functions in server name contexts. Therefore, applying Wolff to the present claims involves nothing but hindsight and the Examiner must be reading into the Wolff reference knowledge whose only source could possibly be Appellant's own disclosure.

Thus, Appellant respectfully submits that Wolff does not teach each and every feature recited in claim 1 as is required under 35 U.S.C. § 102(e). Independent claims 12 and 21 recite similar features in their respective system and computer program product contexts, and thus, Wolff does not teach each and every feature of these claims for similar reasons as noted above with regard to claim 1. At least by virtue of their dependency on claims 1, 12 and 21, respectively, Wolff does not teach each and every feature of dependent claims 2-7, 10, 13-17 and 22. Accordingly, Appellant respectfully requests withdrawal of the rejection of claims 1-7, 10, 12-17, 21 and 22 under 35 U.S.C. § 102(e).

In addition to the above, Wolff does not teach the specific features set forth in dependent claims 2-7, 10, 13-17 and 22. For example, with regard to claims 2, 13 and 22, since Wolff does not teach server name context, Wolff cannot be found to teach that the server name context on the server comprises a set of resources associated with the server name. While Wolff teaches that various servers may handle I/O operations destined for a particular resource, there is nothing in Wolff that teaches that such handling is performed in a server name context or that such a server name context is comprised of a set of resources associated with a server name.

Regarding claims 3 and 14, Wolff does not teach identifying a membership of a resource within the set of resources for the server name context. As noted above, Wolff does not teach a server name context and therefore, cannot teach identifying a membership of a resource within the set of resources for the server name context.

The Final Office Action alleges that this feature is taught at column 6, lines 1-64 and column 7, line 17 to column 8, line 65. However, these sections of Wolff only teach that a single server controls the administrative functions for a resource and that load balancing may be performed in which the administrative functions may be migrated to another server based on the configuration database records for all the volumes and active nodes in the system. There is no teaching, or even mention, of a set of server names, executing a function in a server name context, or identifying a membership of a resource within a set of resources for the server name context.

The Final Office Action has not shown with particularity what the Examiner believes is equivalent to the features recited in the claim. Rather, the Final Office Action points to large sections of the reference and expects Appellant to guess at what the

Examiner intends. Such an approach does not support a rejection based on anticipation under 35 U.S.C. § 102(e).

Thus, in view of the fact that the Final Office Action has not met its burden to show with particularity what elements in Wolff are believed to be the same as the features recited in the claims, and the fact that Wolff in actuality does not teach any of the features of claims 3 and 14, Appellant respectfully asserts that Wolff does not anticipate claims 3 and 14. Accordingly, Appellant respectfully requests withdrawal of the rejection of claims 3 and 14.

As to claims 4 and 15, Wolff does not teach generating a server name tag for the server name, wherein the membership of the resource in the set of resources is identifiable by the server name tag associatively stored with the resource. The Final Office Action again points to the same section of Wolff as discussed above with regard to the rejection of claims 3 and 14 as allegedly teaching this feature. Again, the Final Office Action does not state what the Examiner believes is the same as Appellant's claimed features but rather leaves it to the Appellant to guess at the reasoning of the Examiner. Again, there is nothing in these sections of Wolff, or any other section of Wolff, that teaches a server name tag or identifying membership of a resource in a set of resources by the server name tag. The term "server name tag" is not even used in the Wolff reference and there is no equivalent structure taught in Wolff that could be interpreted to be a server name tag. The Final Office Action has not met its burden to show anticipation by the Wolff reference. Therefore, Appellant respectfully requests withdrawal of the rejection of claims 4 and 15.

With regard to claims 5 and 16, Wolff does not teach a server name tag being generated based on a value of the server name and a value derived from a data structure that stores a server name. The Final Office Action alleges that this feature is taught in Wolff in Figure 1B, column 5, line 56 to column 8, line 24 and column 15, line 61 to column 16, line 45. However, these sections of the Wolff reference do not make any mention whatsoever of a server name tag or the server name tag being generated based on a value of a server name and a value derived from a data structure that stores a server name. Moreover, these sections of Wolff do not describe any structure that could even arguably be interpreted to be equivalent to the features recited in claims 5 and 16. Yet again, the Final Office Action merely points to large sections of a reference and alleges that its taught somewhere in these sections. None of these sections have anything to do with the features recited in claims 5

and 16. Accordingly, Appellant respectfully requests withdrawal of the rejection of claims 5 and 16 under 35 U.S.C. § 102(e).

As to claims 6 and 17, Wolff does not teach that a value derived from the data structure is a position value of the server name within a server name table that stores the set of server names. The Final Office Action alleges that this feature is taught in Figures 5A-B, column 19, line 32 to column 21, line 33 and column 23, lines 12-54. Again, the Final Office Action references large sections of the Wolff reference without any guidance as to what the Examiner may be considering the same as the features of the present claim. While these sections may illustrate node names, there is no teaching or suggestion as to how such node names are derived. Thus, Wolff does not teach a value being derived from a data structure wherein the value is a position value of the server name within a server name table. These sections of Wolff do not even mention a server name table let alone deriving a value from a position value of a server name within the server name table. There is nothing in Wolff that can even arguably be interpreted to teach this feature. Therefore, Appellant respectfully requests withdrawal of the rejection of claims 6 and 17 under 35 U.S.C. § 102(e).

With regard to claim 10, it should be noted that claim 10 is dependent on claim 9 which has been indicated by the Examiner to contain allowable subject matter (see Advisory Action mailed June 14, 2002. Thus, claim 10 should be indicated as containing allowable subject matter as well. However, in addition to be dependent on claim 9, claim 10 recites repeatedly identifying a plurality of resources that are applicable to the server name by searching a plurality of resource data structures for matching server name masks. Wolff does not teach or suggest such a feature. Wolff does not teach server name masks, let alone searching a plurality of resource data structures for matching server name masks. Moreover, Wolff does not teach repeatedly performing such searching to repeatedly identity a plurality of resources that are applicable to a server name. Thus, Appellant respectfully requests withdrawal of the rejection of claim 10 under 35 U.S.C. § 102(e).

In response to Appellant's arguments with regard to the dependent claims above, the Examiner merely states in the Final Office Action that "Claims 2-11, 13-20 and 22 are rejected at least by virtue of their dependency on independent claims 1, 12 and 21 and by other reasons set forth in the previous Office Action (paper #3, mailed on 8/28/2001)"

(emphasis added). Again, it appears that the Examiner is confused as to the requirements for establishing a case of anticipation under 35 U.S.C. § 102(e). Dependent claims cannot be rejected simply because they are dependent. The Examiner must examine each claim on its merits and cannot simply "wave his hand" at the specific features recited in the dependent claims. If the Examiner cannot find each and every feature of every dependent claim identically in the cited reference, the Examiner must withdraw the rejection of those claims under 35 U.S.C. § 102(e).

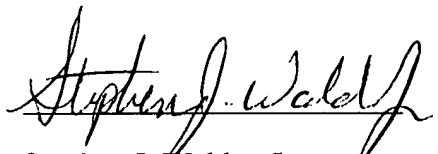
Moreover, the Examiner's referring to the "reasons set forth in the previous Office Action" does not rebut Appellant's specific arguments presented above. Merely referring back to the original rejection does not address why the Examiner believes that Appellant's arguments are not persuasive when Appellant provides very detailed reasoning as to why the very sections cited in the original rejection do not teach what is alleged in the Office Action. Therefore, Appellant respectfully submits that the Final Office Action does not set forth a persuasive case of anticipation in view of the clear and convincing analysis of Wolff provided by Appellant.

In summary, the Wolff reference does not meet the standard of every element of the claimed invention being identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). Moreover, the Final Office Action has not met the burden of setting forth how the Wolff et al. reference allegedly teaches all of the features of claims 1-7, 10, 12-17, 21 and 22. Appellant respectfully urges that the Examiner must show with particularity what elements in the reference the Examiner considers the same as the features recited in the claim and illustrate how these elements are arranged in the same manner as the features of the presently claimed invention. The Examiner has not done this in the rejection of any of the claims in the Final Office Action. This is because the Wolff reference, in actuality, does not teach any of the features of the pending claims. Appellant therefore respectfully requests withdrawal of the rejection of all of claims 1-7, 10, 12-17, 21 and 22 under 35 U.S.C. § 102(e).

CONCLUSION

In view of the above, Appellant respectfully submits that all of claims 1-22 are allowable over the cited prior art and that the application is in condition for allowance. Accordingly, Appellant respectfully requests the Board of Patent Appeals and Interferences to overturn the rejections set forth in the Final Office Action.

Respectfully submitted,



Stephen J. Walder, Jr.
Reg. No. 41,534
Carstens, Yee & Cahoon, LLP
PO Box 802334
Dallas, TX 75380
(972) 367-2001
Attorney for Appellant

APPENDIX OF CLAIMS

The text of the claims involved in the appeal are:

1. A method for executing a function on a server in a distributed data processing system, the method comprising the computer-implemented steps of:

receiving a request for a function, wherein the request comprises an input specifying a server name, wherein the server responds to requests directed to a set of server names; and

executing the function in a server name context on the server as directed by the input specifying the server name.

2. The method of claim 1 wherein the server name context on the server comprises a set of resources associated with a server name.

3. The method of claim 2 further comprising identifying a membership of a resource within the set of resources for the server name context.

4. The method of claim 3 further comprising generating a server name tag for the server name, wherein the membership of the resource in the set of resources is identifiable by the server name tag associatively stored with the resource.

5. The method of claim 4 wherein the server name tag is generated based on a value of the server name and a value derived from a data structure that stores the server name.

6. The method of claim 5 wherein the value derived from the data structure is a position value of the server name within a server name table that stores the set of server names.

7. The method of claim 1 wherein the request for the function is received from a network.

10. The method of claim 9 further comprising:

repeatedly identifying a plurality of resources that are applicable to the server name by searching a plurality of resource data structures for matching server name masks.

12. A data processing system comprising:

means for receiving a request for a function, wherein the request comprises an input specifying a server name, wherein the server responds to requests directed to a set of server names; and

means for executing the function in a server name context on the server as specified by the input containing the server name.

13. The data processing system of claim 12 wherein the server name context on the server comprises a set of resources associated with a server name.

14. The data processing system of claim 13 further comprising identification means for identifying a membership of a resource within the set of resources for the server name context.

15. The data processing system of claim 14 further comprising generation means for generating a server name tag for the server name, wherein the membership of the resource in the set of resources is identifiable by the server name tag associatively stored with the resource.

16. The data processing system of claim 15 wherein the server name tag is generated based on a value of the server name and a value derived from a data structure that stores the server name.

17. The data processing system of claim 16 wherein the value derived from the data structure is a position value of the server name within a server name table that stores the set of server names.

21. A computer program product on a computer readable medium for use in a data processing system, the computer program product comprising:

first instructions for receiving a request for a function, wherein the request comprises an input specifying a server name, wherein the server responds to requests directed to a set of server names; and

second instructions for executing the function in a server name context on the server as specified by the input containing the server name.

22. The computer program product of claim 21 wherein the server name context on the server comprises a set of resources associated with a server name.